TP8.00026

Modifying COGENT to Study Snowflake Divertors

R.H. Cohen[1,2], M. Dorf[2], M. Dorr[2], D. D. Ryutov[2], P. Schwartz[3]

[1]CompX, Del Mar, CA 92014

[2]Lawrence Livermore National Laboratory, Livermore, CA 94550

[3]Lawrence Berkeley National Laboratory, Berkeley, CA 94720

APS-DPP Meeting, New Orleans LA, Oct 27-31, 2014

This work was performed for USDOE, at LLNL under contract DE-AC52-07NA27344, at CompX under sub-contract from LLNL, and at LBNL under contract DE-AC02-05CH11231

ABSTRACT

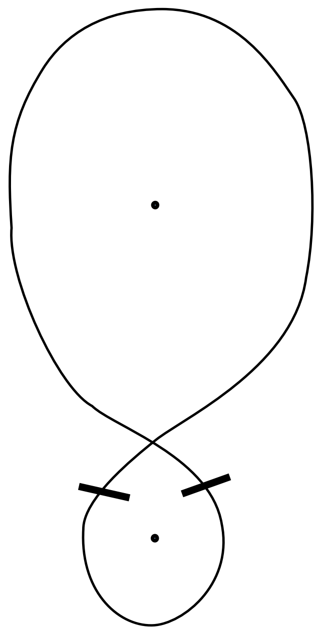
The snowflake divertor concept entails modifying the poloidal field system of a tokamak to produce a 2nd-order null in magnetic-field strength in place of the conventional 1st-order null x point within the equilibrium magnetic-field separatrix. It more effectively spreads the divertor heat load and offers a number of other advantages. We describe plans to modify the COGENT edge kinetic code to study snowflake divertors. COGENT employs mapped multi-block grid technology to handle the geometric complexity of the conventional divertor configuration. To simulate snowflake divertors, the number of grid blocks is increased from 8 to 12, consistent with the modified topology of the exact snowflake configuration. We examine the applicability of the modified structure to study configurations that are not exactly snowflakes, the so-called “snowflake-plus” and “snowflake-minus” configurations. Initial applications of the modified code will be assessment of collisionless orbit dynamics and neoclassical transport.

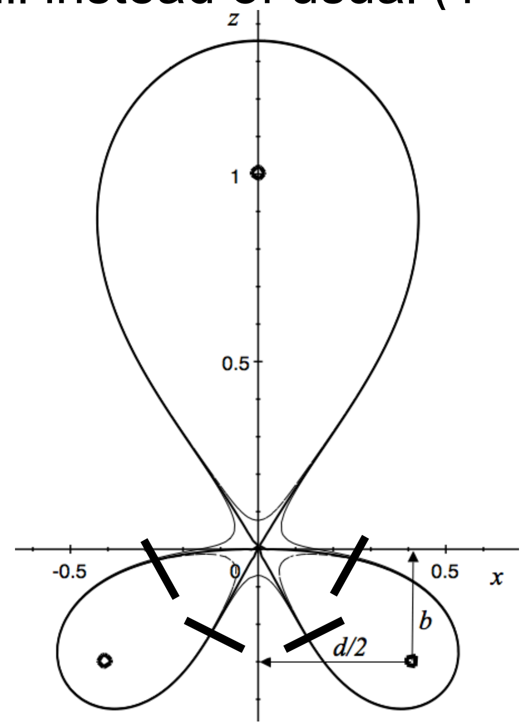
OUTLINE

- What and why of snowflake divertors
- Exact versus approximate snowflake divertors
- Objectives for modeling snowflakes with COGENT
- COGENT gridding strategy for conventional divertor tokamaks
- Strategy for extension to snowflakes: simple!
- First step: model and test local region about poloidal field null

Snowflake divertors: What and Why

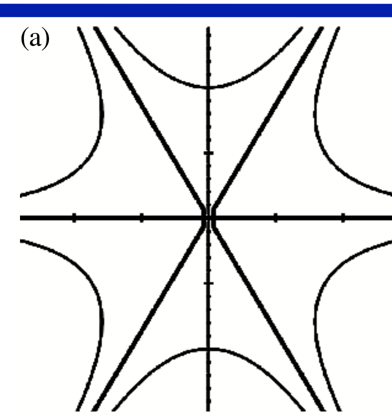
- What: Extra coil(s) to produce 2nd-order null instead of usual (1st-order) x point in SOL.



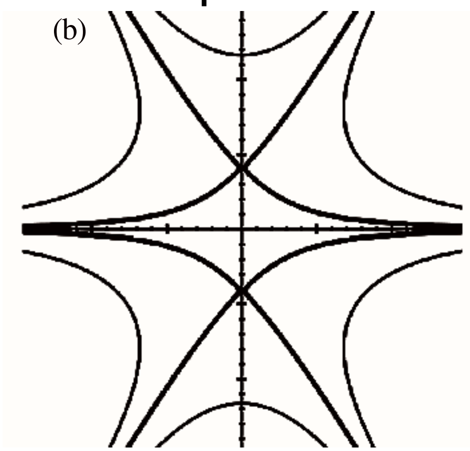


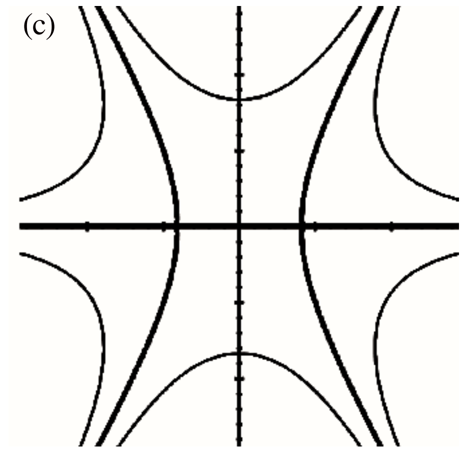
- Why:
  - Primary benefit, spreading of heat load via increased flux expansion.
  - Secondary benefits: further spreading among multiple divertor legs via MHD convection
  - Further isolation of main SOL and divertor legs RE instabilities (increased shear)
  - Other benefits, e.g. reduced peak heat load during ELMs

Exact versus approximate snowflake divertors

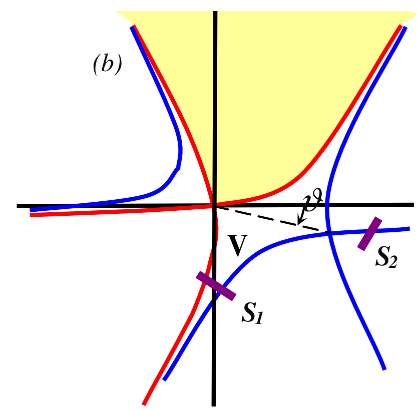


- Exact snowflake: perfect tuning of coils to achieve 2nd-order null
- Structurally unstable: if one of the coils has current a bit too high or low, the 2nd-order null splits into 2 nearby 1st-order nulls
- Snowflake plus: Snowflake minus:





- Above examples are symmetric approximate snowflakes. They needn't be. e.g.:



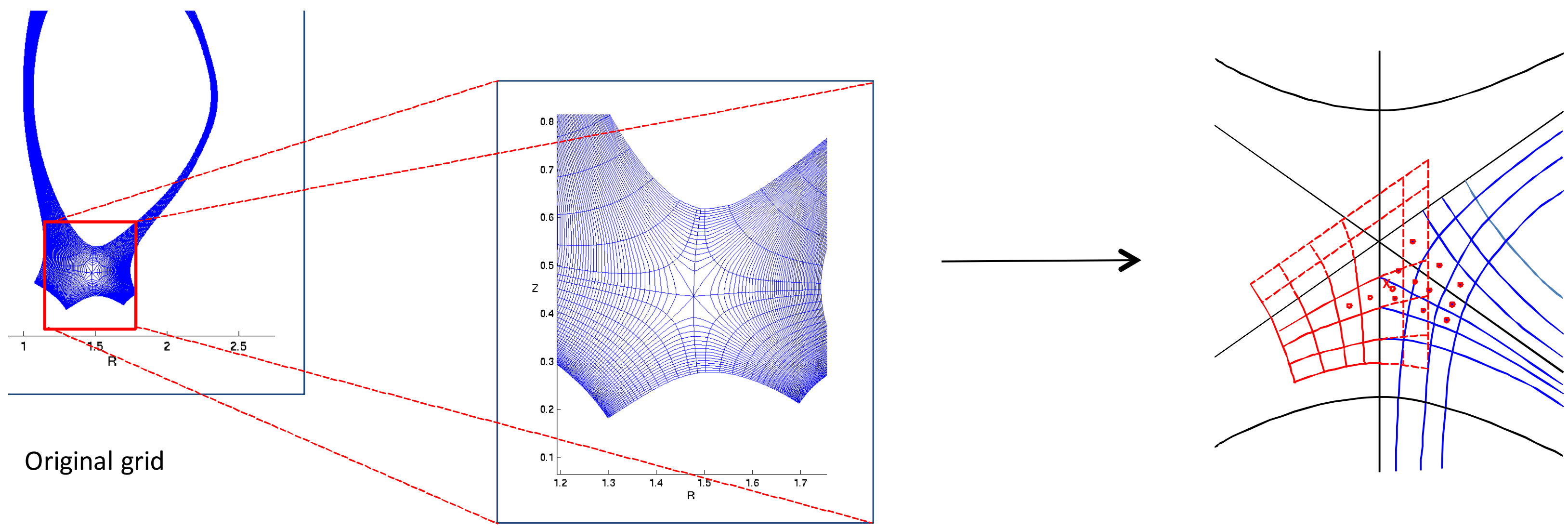
- If the 1st-order nulls are close enough, macroscopic behavior mostly indistinguishable from exact snowflake.

Objectives for modeling snowflakes with COGENT

- Snowflake divertors are getting a lot of attention at DIII-D and elsewhere, need to model them.
- Initial objectives similar to those for conventional divertors:
  - Neoclassically driven flows and radial transport in presence of divertor losses
  - Distribution of collision-driven losses to divertor plates
- Have divertor geometry in the mix as COGENT capability is expanded (e.g. to include 5-D physics)

COGENT gridding strategy for conventional divertors: abandon field-line following near x point

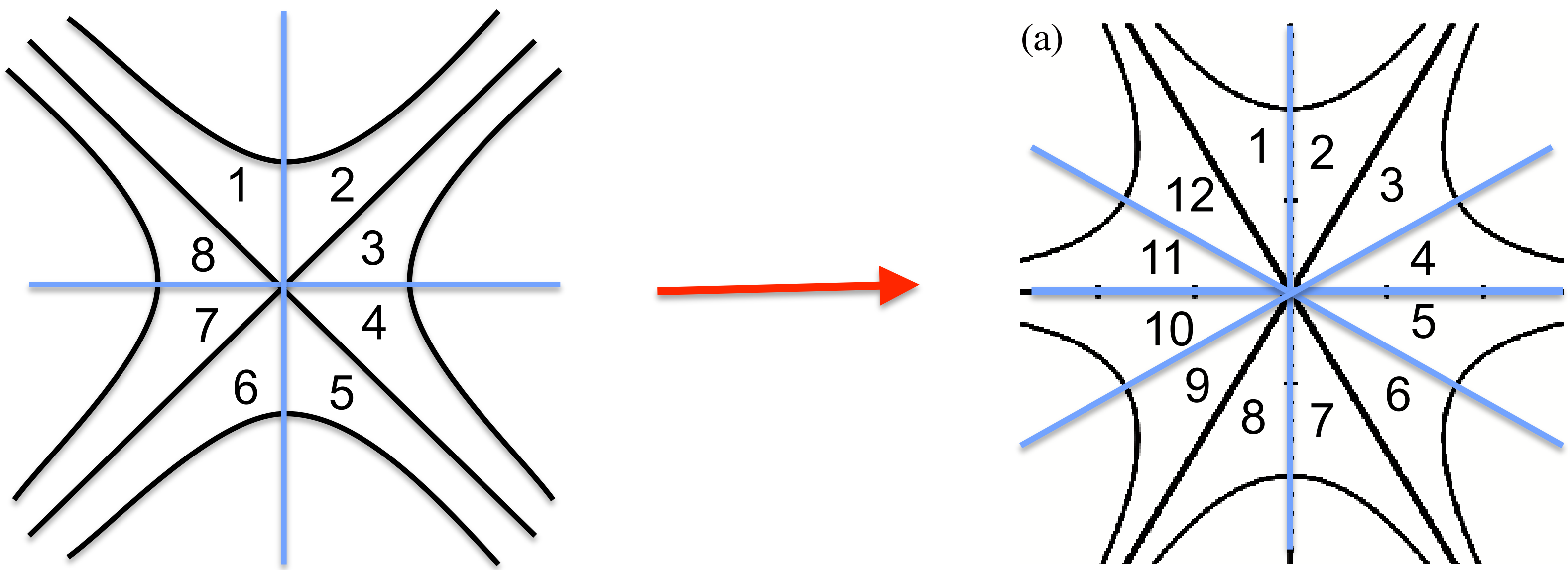
- When the divertor version of COGENT was first developed it was noted that the nominally 4th-order discretization was yielding results for advection converging more slowly than $(\Delta x)^4$
  - Explanation: curvature, metrics becoming singular as x point is approached.
- Solution: Gridding that follows flux surfaces away from x point but departs so as to preserve smoothness near x point
  - Flows near x point not flux-surface-following anyway
  - Use 4th-order interpolation to fill ghost cells



Original grid

Strategy for extension to snowflakes: simple!

- Implication of extrapolated grid strategy for snowflake divertor: Since field-line following is abandoned anyway, a single grid structure generated for an exact snowflake divertor geometry is likely to work for nearby approximate snowflakes
- Main complication: increase of number of grid blocks required to describe region about field null increases from 8 to 12:



First step: model local region about poloidal field null

- Ryutov et al PPCF '08: cubic expansion of flux surface about null.
  - Neglecting current near null, have flux function

$$\Phi = l_1 x + l_2 z - q_3 x^2 + 2q_2 xz + q_3 z^2 + c_1 x^3 - 3c_4 x^2 z - 3c_1 xz^2 + c_4 z^3$$

  - And fields

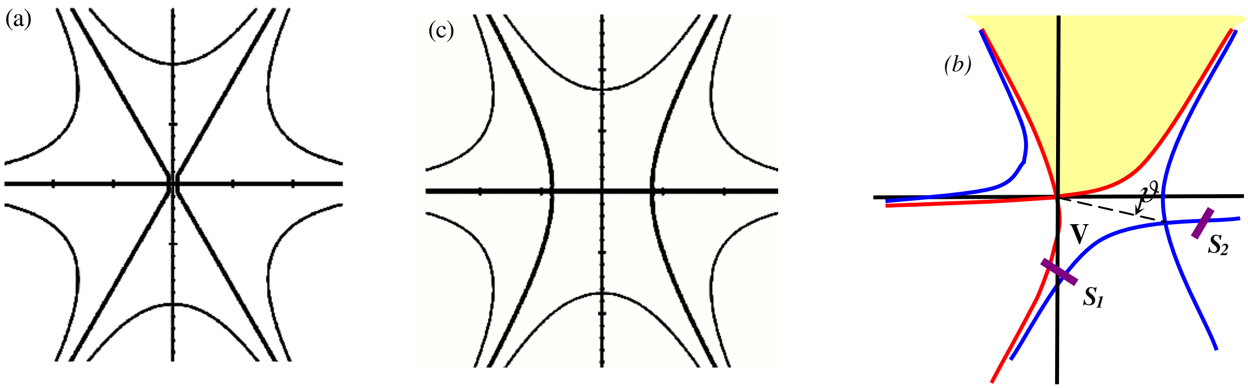
$$-(R+x)B_x = l_2 + 2q_2 x + 2q_3 z - 3c_4(x^2 - z^2) - 6c_1 xz,$$

$$(R+x)B_z = l_1 - 2q_3 x + 2q_2 z + 3c_1(x^2 - z^2) - 6c_4 xz.$$

- With suitable choices of coefficients, can make exact snowflake and approximate snowflakes
- Strategy:
  - Starting from exact snowflake coefficients, generate extrapolated grid as discussed above
  - Do runs with **B** on this grid evaluated for exact and approximate snowflakes, compare physics results (next slide)

REMARKS

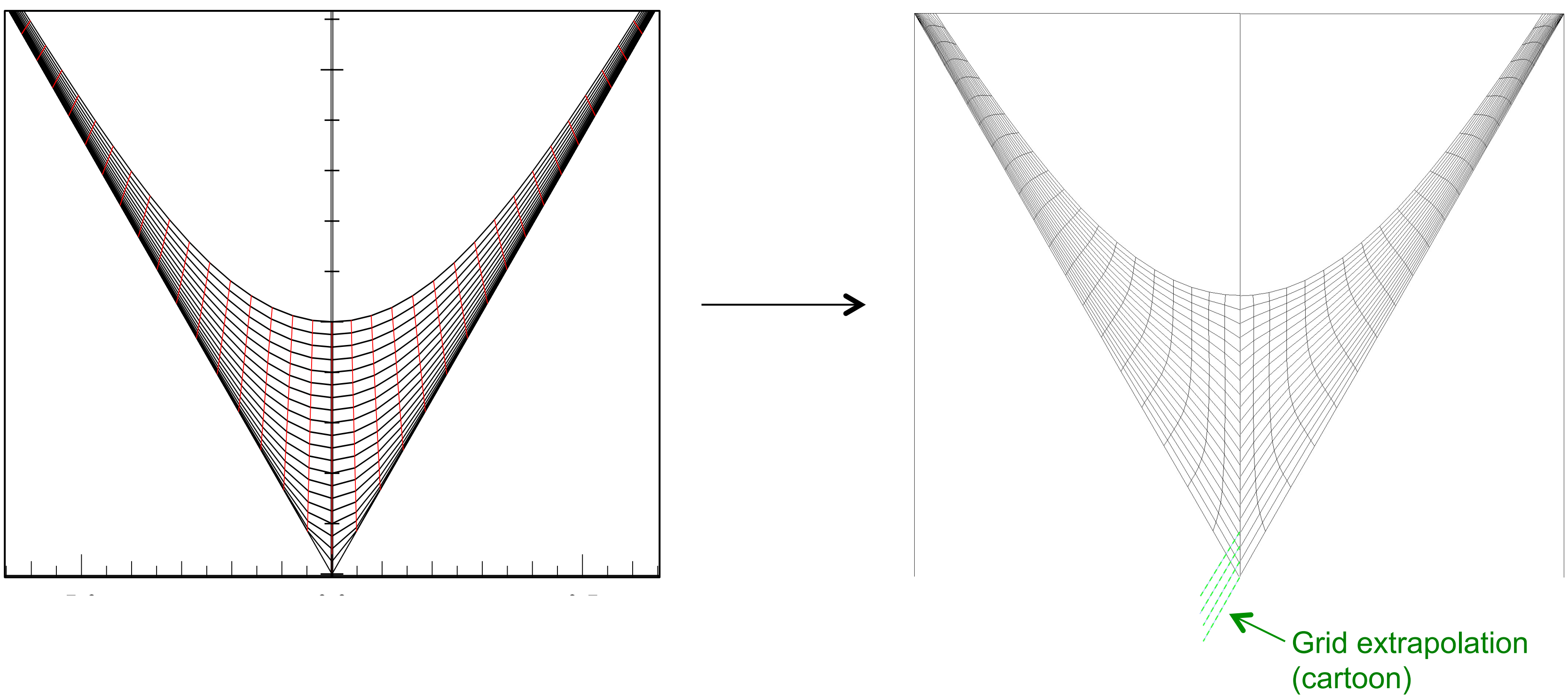
- We use grid generator developed by P. Schwartz, which does variational optimization of grid smoothness and field line following with weights that vary with field-line curvature (builds on approach of Brackbill and Salzman, JCP 1982)
- Initial testing will be with pure advection (no collisions): Initiate (half) Maxwellian in main SOL; predictable difference of fluxes on various divertor plates depending on type of approximate snowflake



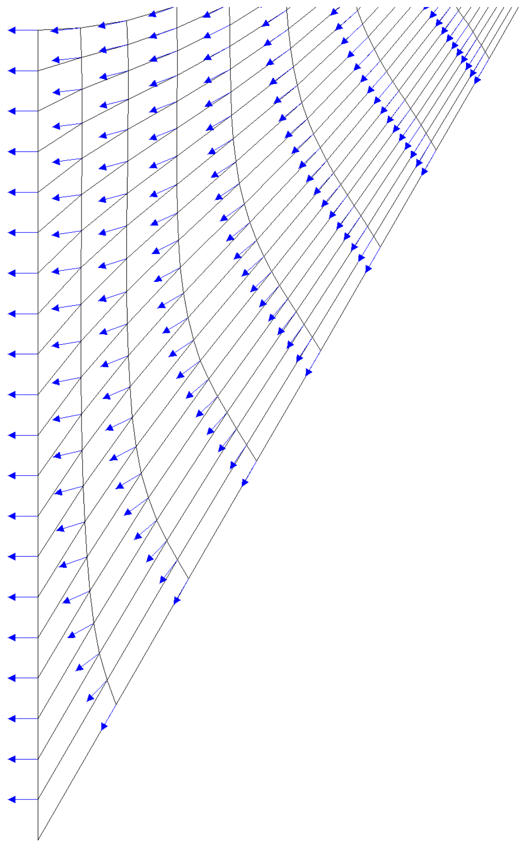
- Subsequent studies: add collisions (neoclassical); add model of MHD convective mixing near null; full SOL. Compare with analytic models that may be available, and with experiments

First step completed: Generating smooth extrapolatable grid for snowflake divertor region

- Provided input to the optimizer: a grid that follows flux surfaces (but has sharp curves near separatrix).
- Output obtained from optimizer, follows field lines away from null, smooth/straighter near null



Plot of field vectors illustrates what the optimizer does



Grids follow flux surfaces where their curvature is weak (away from field null), departs where curvature is strong